

fig. 2
(PRIOR ART)

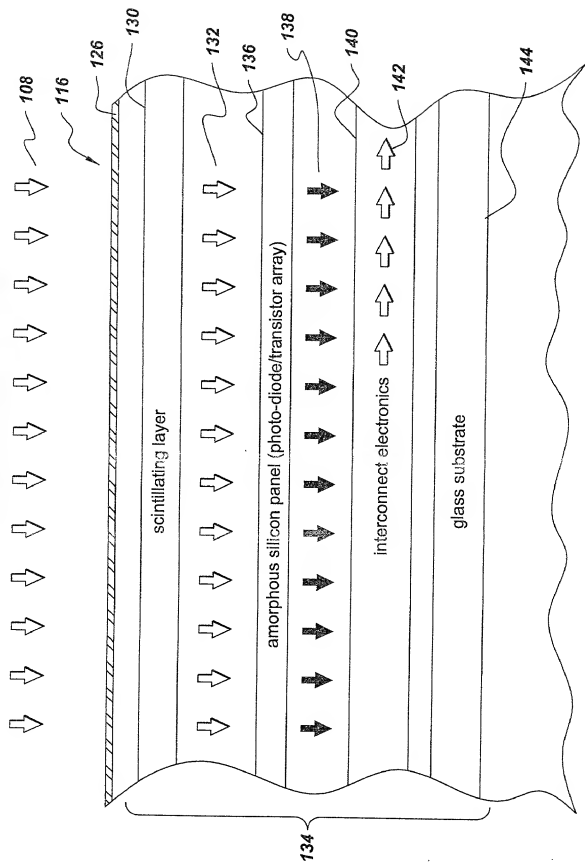
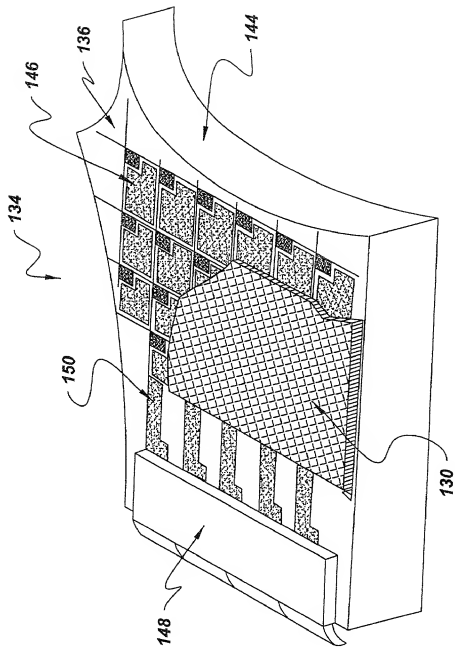


fig. 3

fig. 4
(Prior Art)



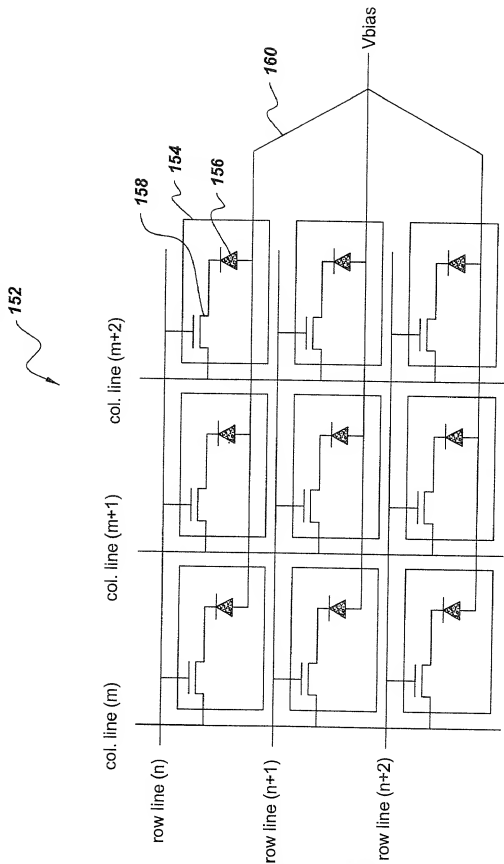


fig. 5
(Prior Art)

fig. 6
(PRIOR ART)

7/53

FLAT PANEL DETECTOR

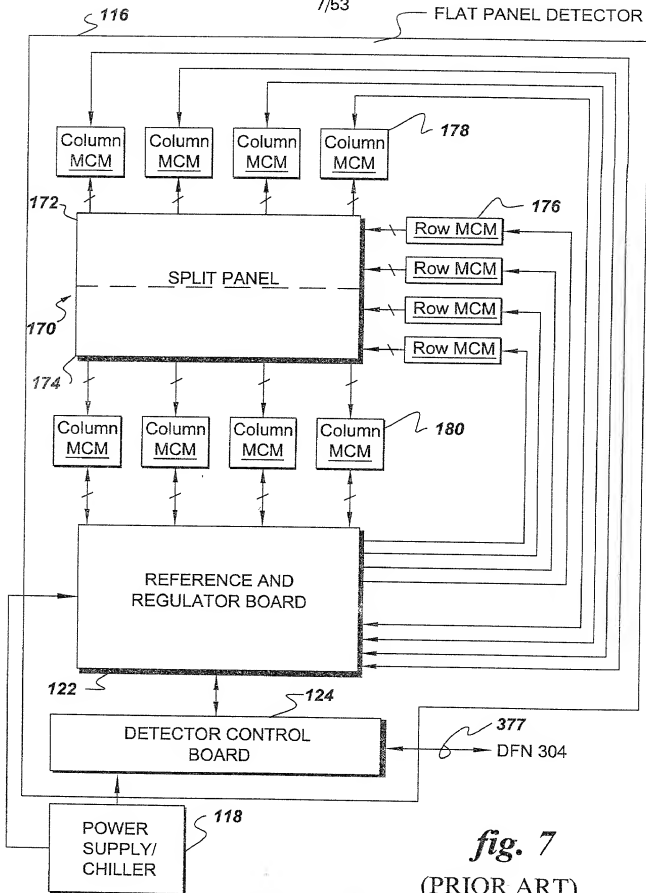
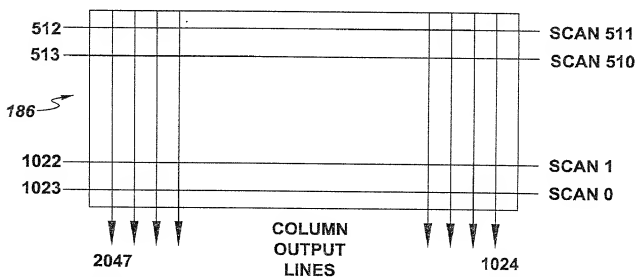
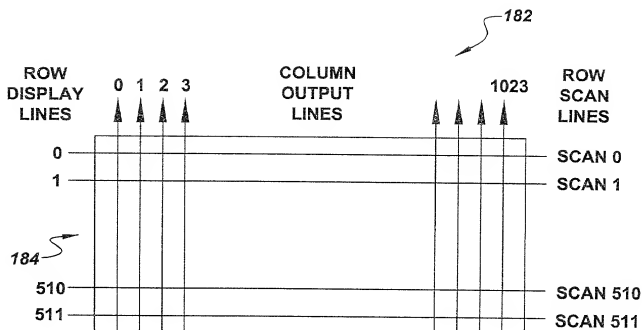


fig. 7
(PRIOR ART)



CARDIAC/SURGICAL DIGITAL X-RAY PANEL

fig. 8
(PRIOR ART)

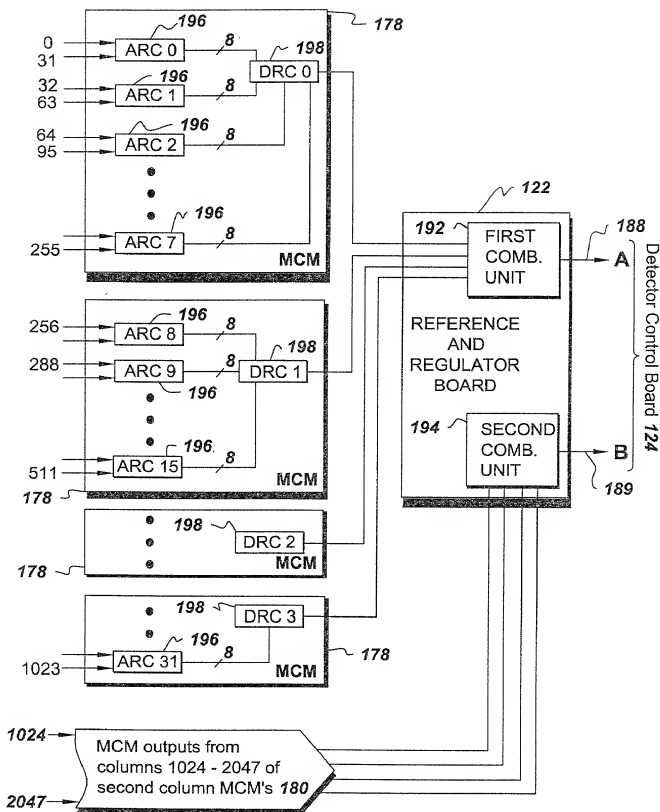
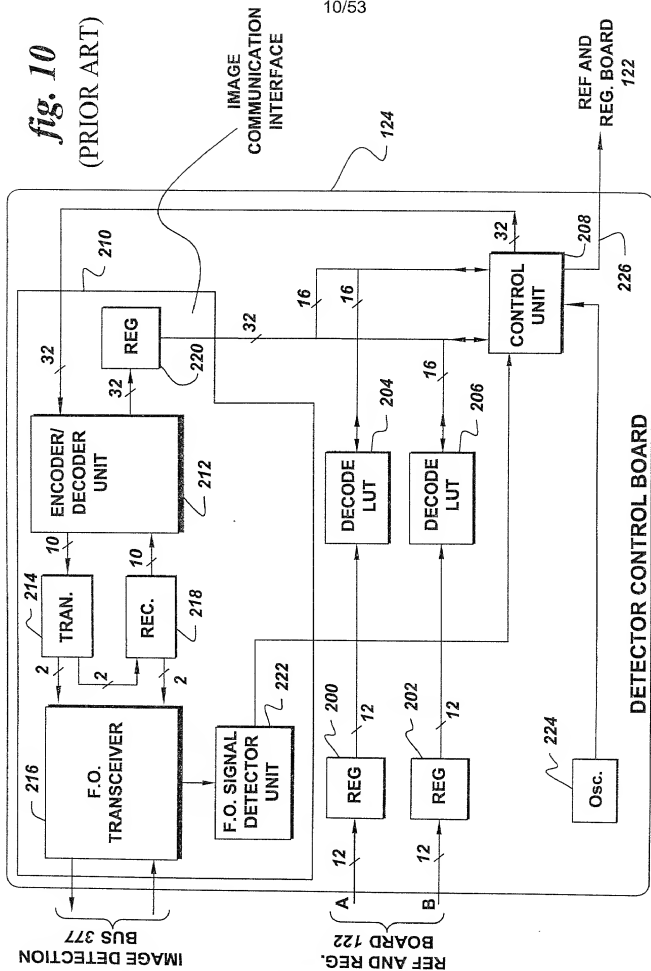
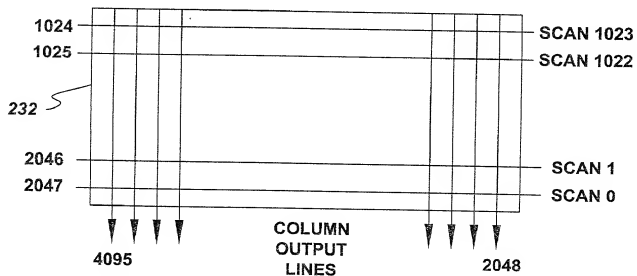
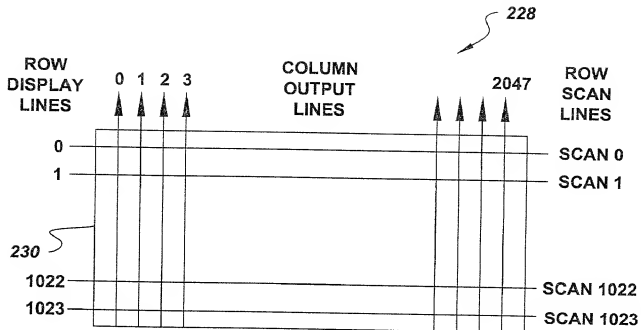


fig. 9 (PRIOR ART)

IMAGE DETECTION
BUS 377

fig. 10
(PRIOR ART)





RADIOGRAPHY DIGITAL X-RAY PANEL

fig. 11 (PRIOR ART)

0977549 050766

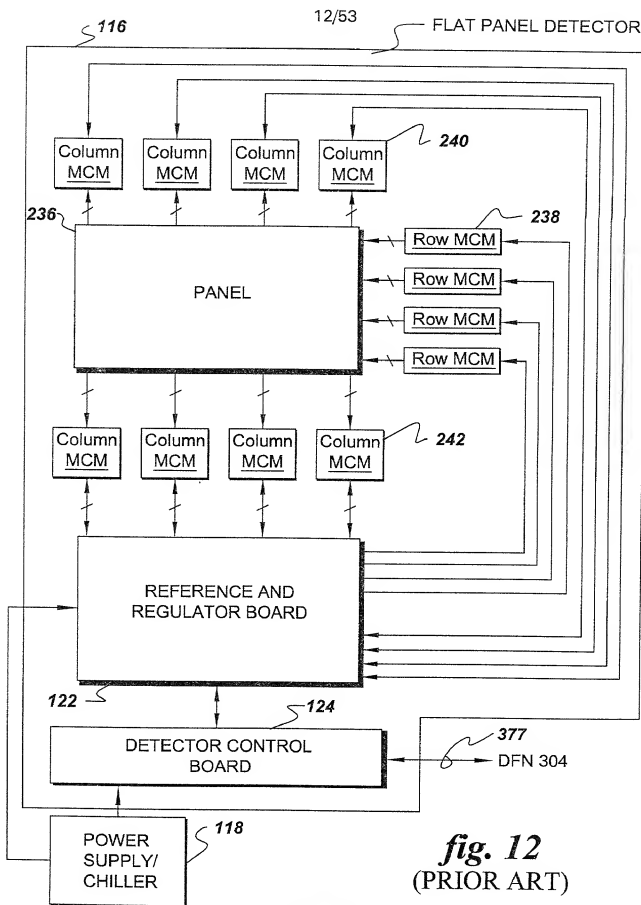
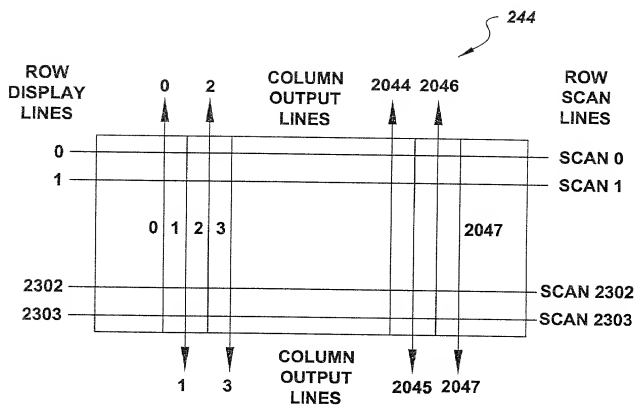


fig. 12
(PRIOR ART)



MAMMOGRAPHY DIGITAL X-RAY PANEL

fig. 13
(PRIOR ART)

14/53

FLAT PANEL DETECTOR

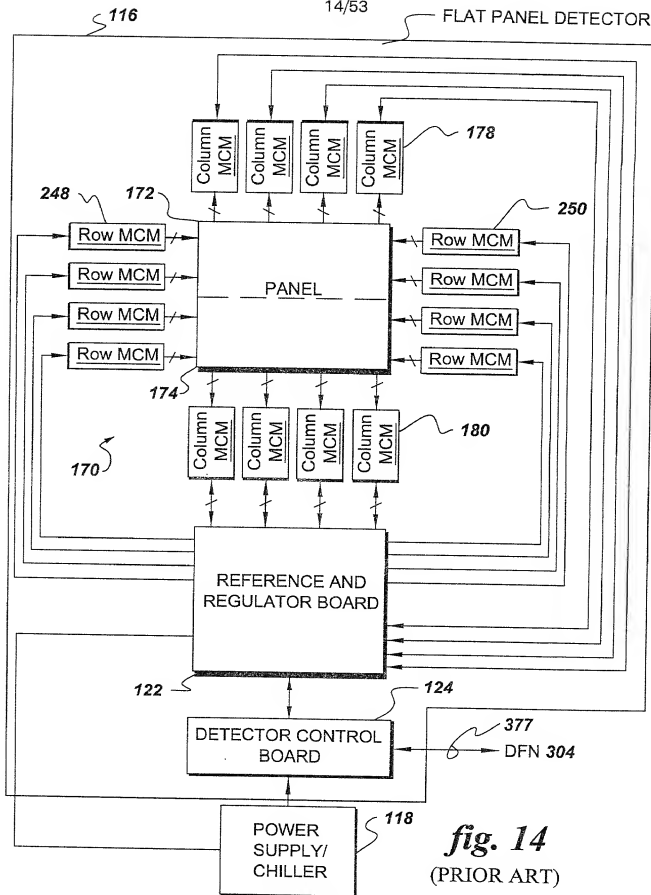


fig. 14
(PRIOR ART)

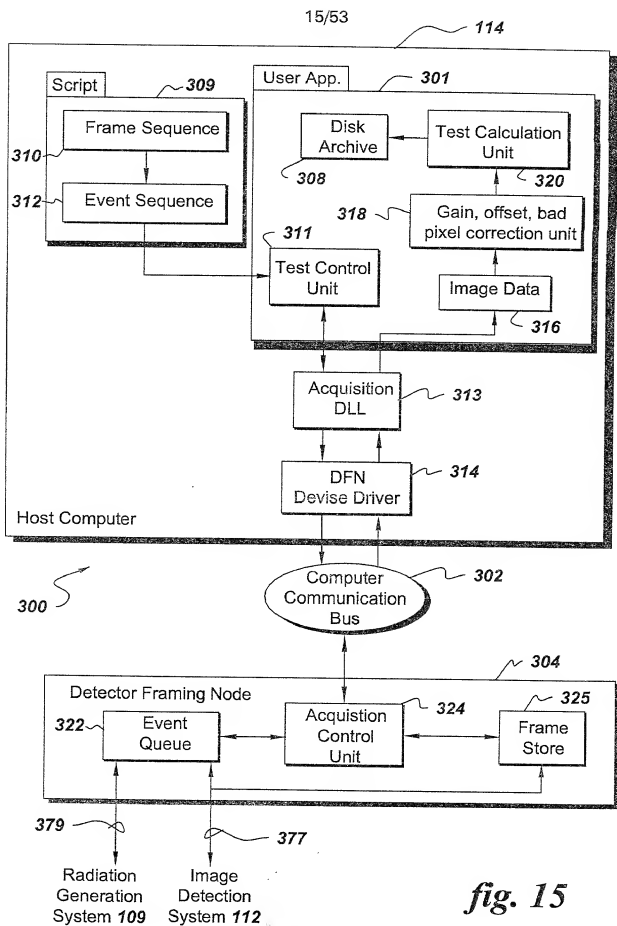


fig. 15

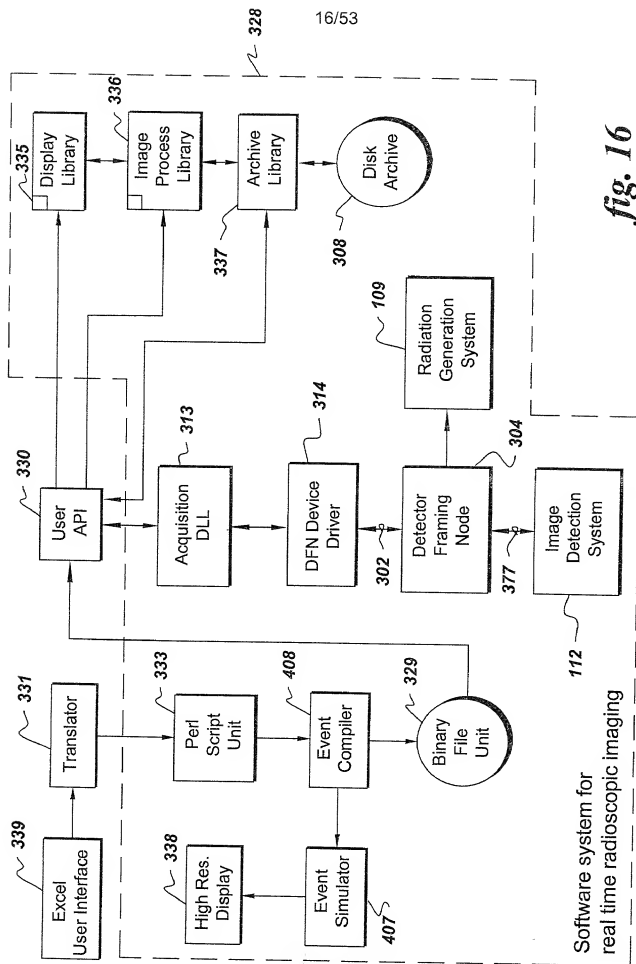


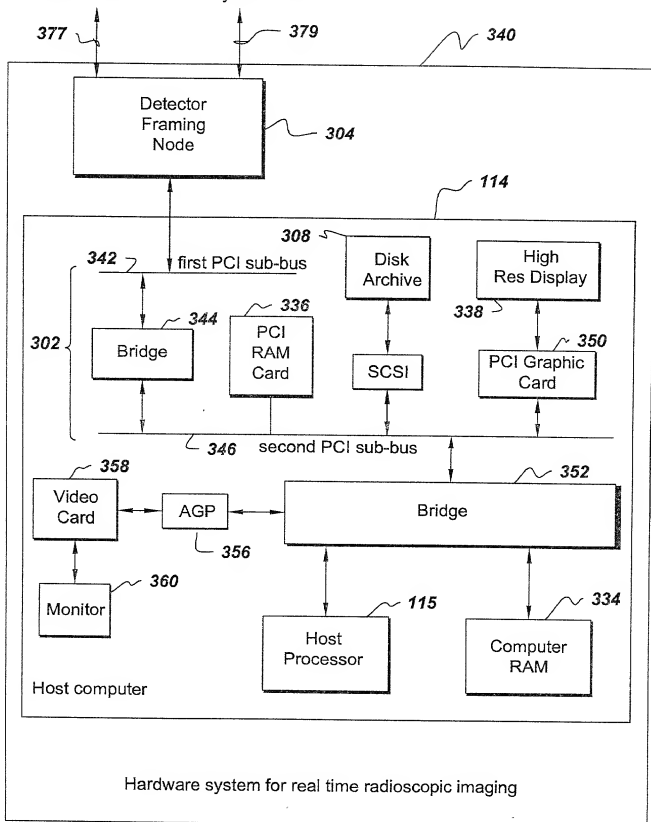
fig. 16

17/53

Image Detection
System 112

Radiation Generation
System 109

fig. 17



Hardware system for real time radiosopic imaging

102050-01/2/2000

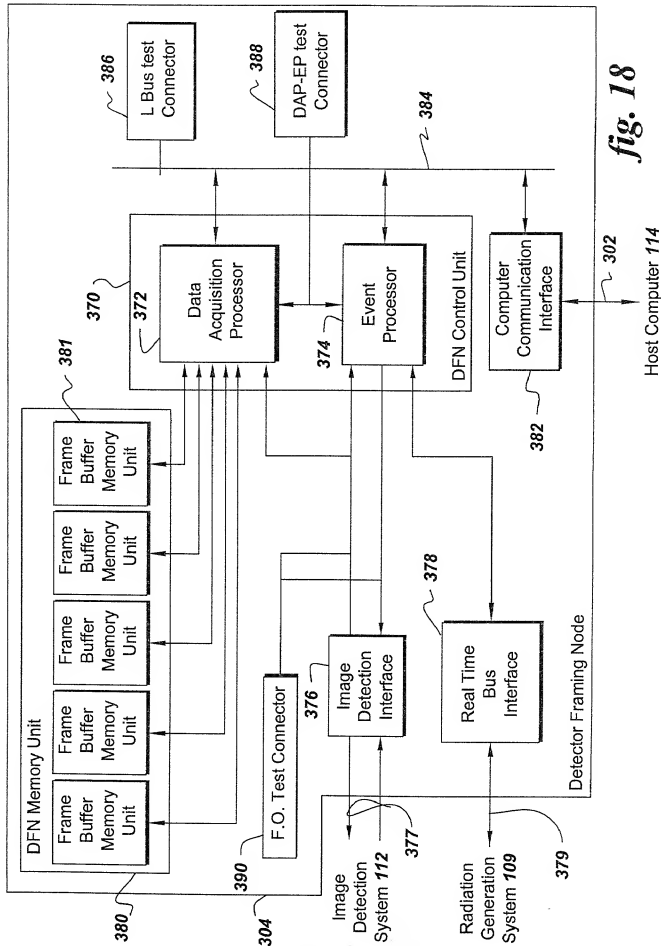


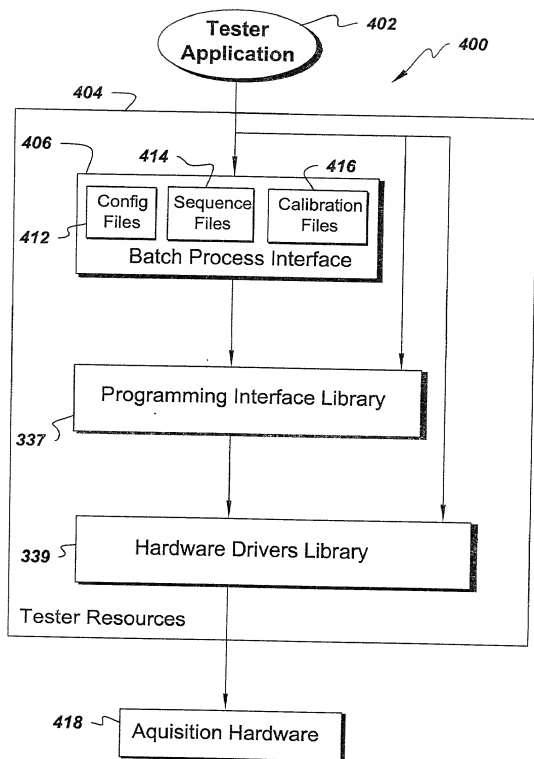
fig. 18

	(fm/sec)	length	latency	memory	offset	gbr
Panel Setup	Real Time	30	unlimited	host	none	
Single Frame	Post Process	-	<5 frames Delay ~.1 sec	"	y	
Single Frame	Post Process	-	Delay ~.2 sec	"	y	y
Real Time	Real Time	R	unlimited	host	none	
Real Time	Real Time	R-X	unlimited	"	y	
Real Time	Real Time	R-Y	unlimited	"	y	y

fig. 19

Modality	Image size	Frames Stored host memory
Cardiac	1024 X 1024	200
Rad	2048 X 2048	50
Mammo	2304 X 2048	44

fig. 20

*fig. 21*

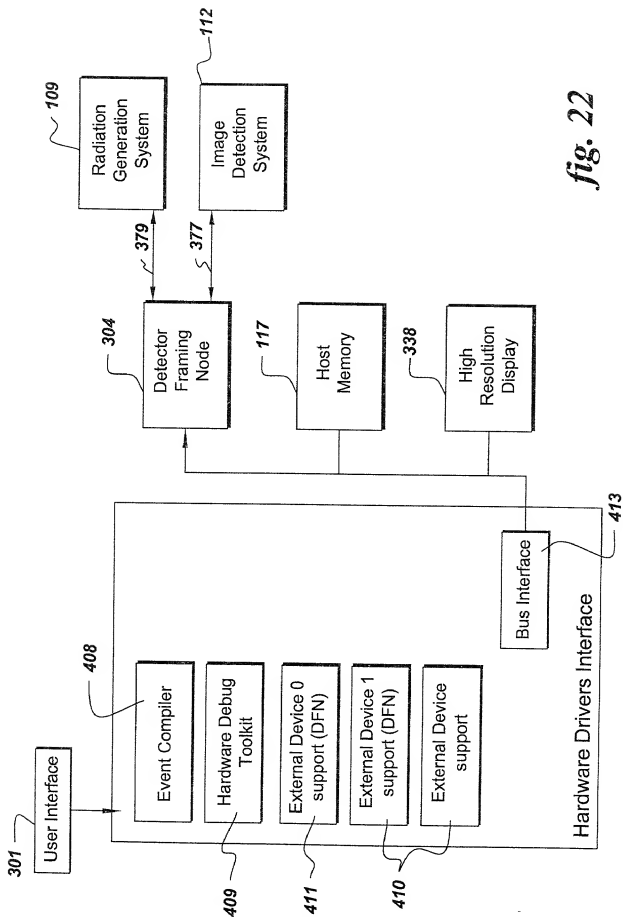


fig. 22

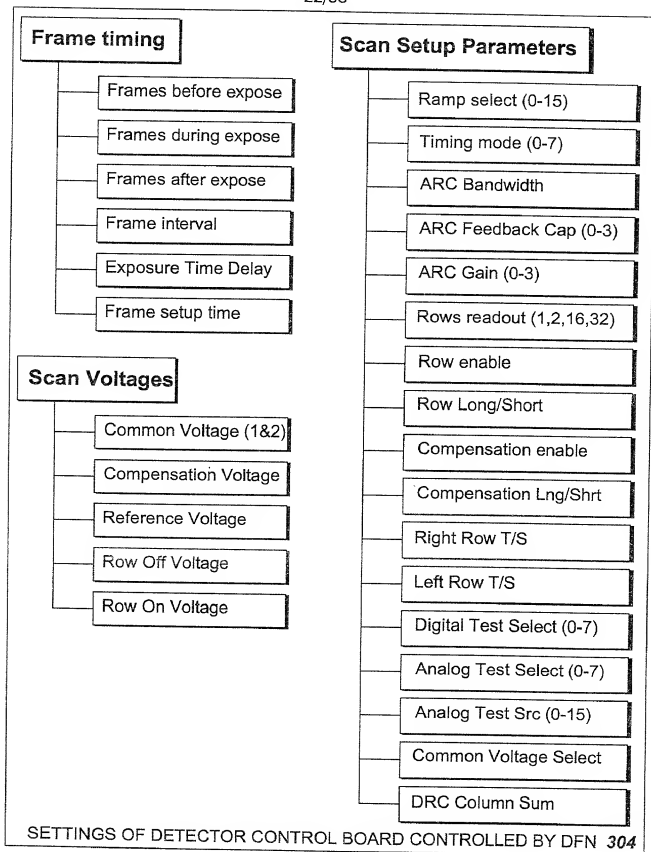
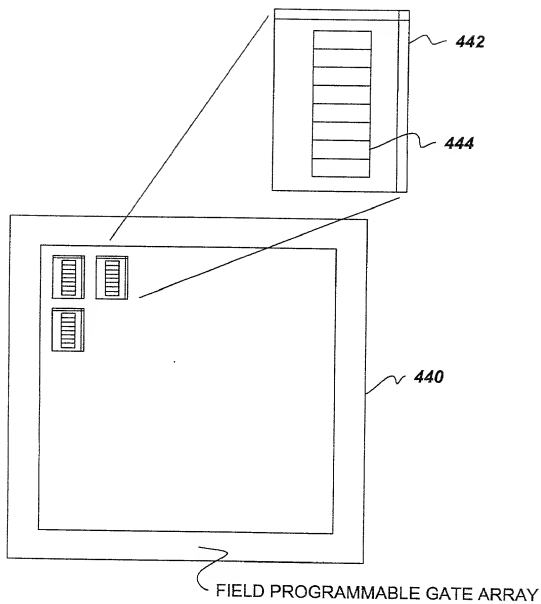


fig. 23

*fig. 24*

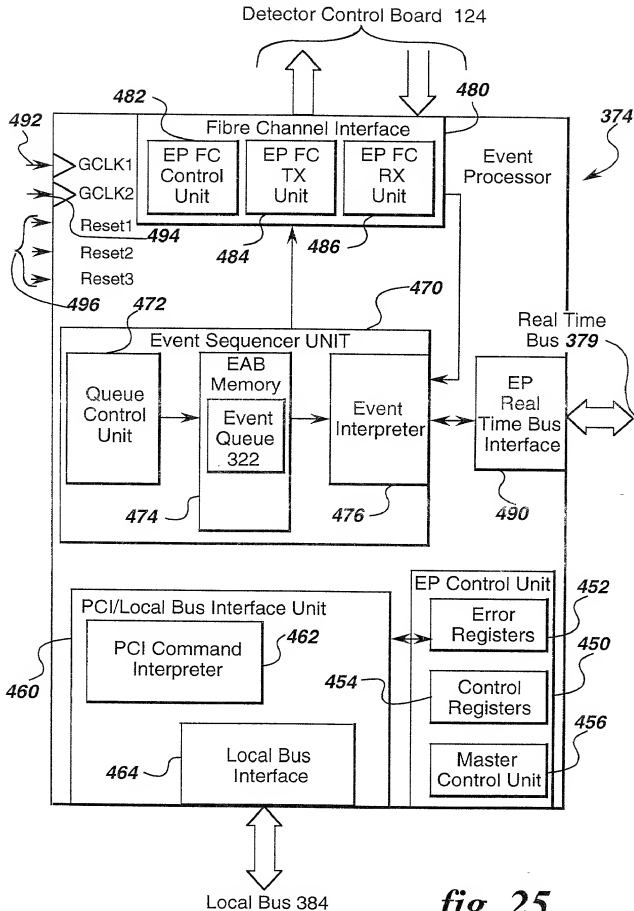


fig. 25

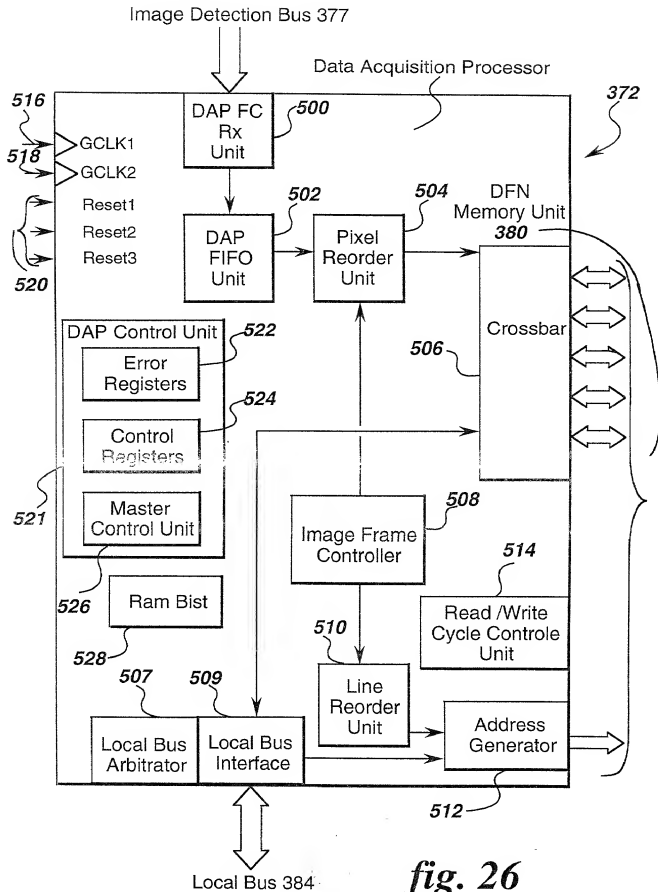


fig. 26

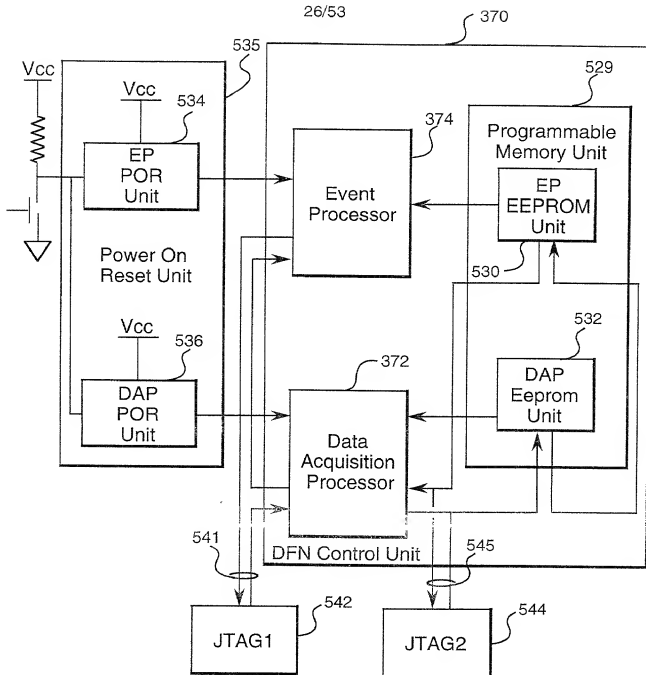


fig. 27

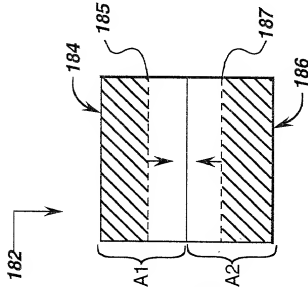


fig. 28

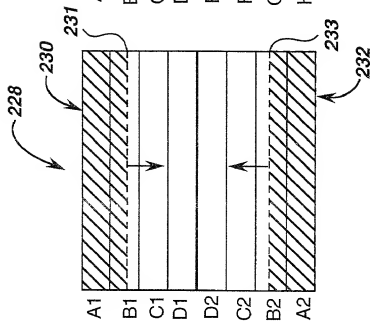


fig. 29

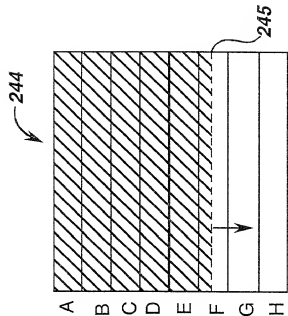
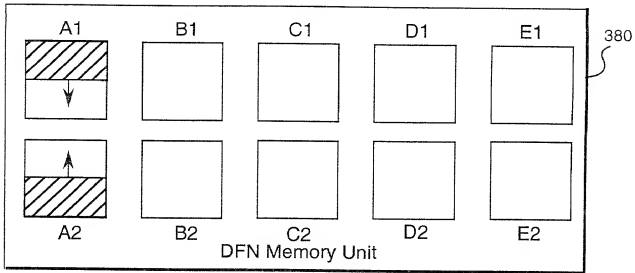
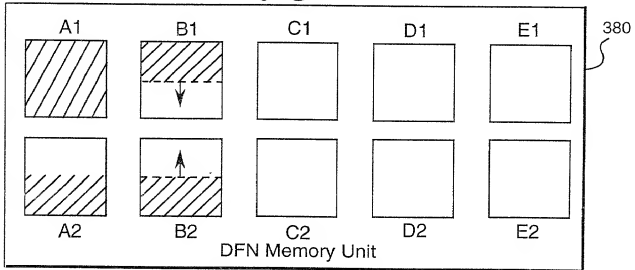
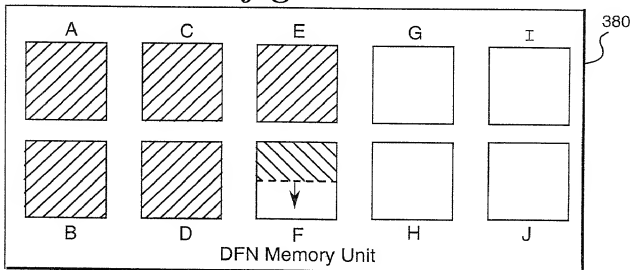


fig. 30

*fig. 31**fig. 32**fig. 33*00774510, 050704
10/20/2009 09:42:40

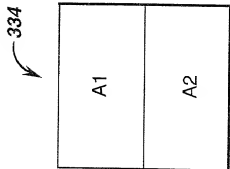


fig. 34

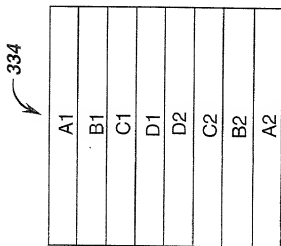


fig. 35

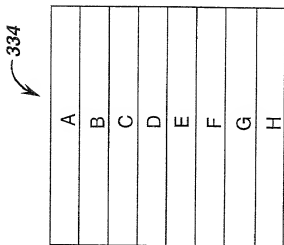


fig. 36

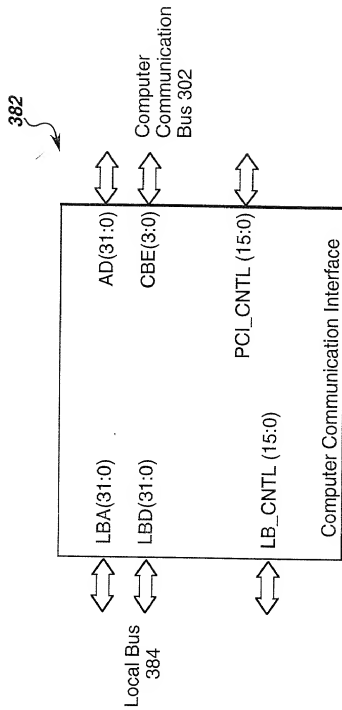


fig. 37

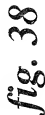
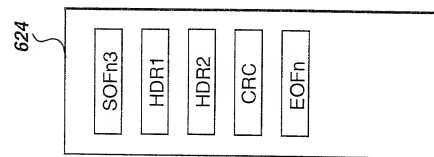
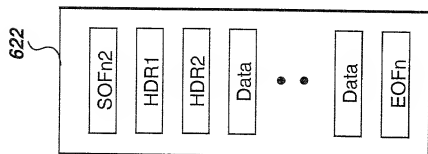
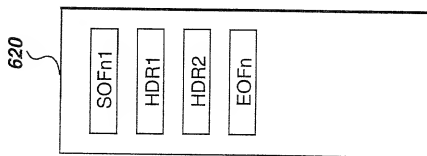


fig. 38

*fig. 41**fig. 40**fig. 39*

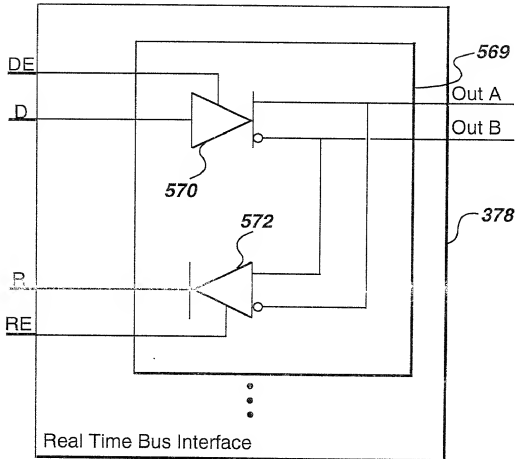
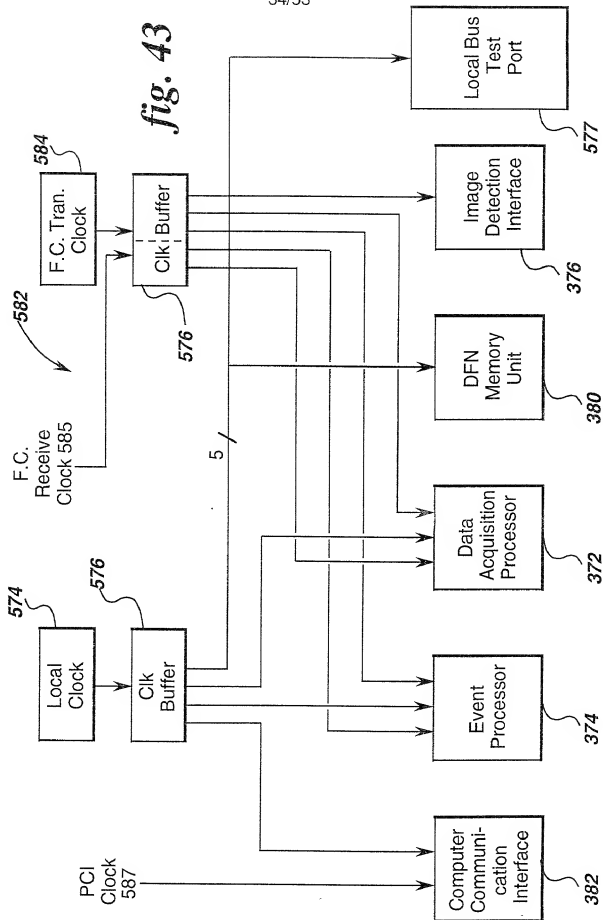
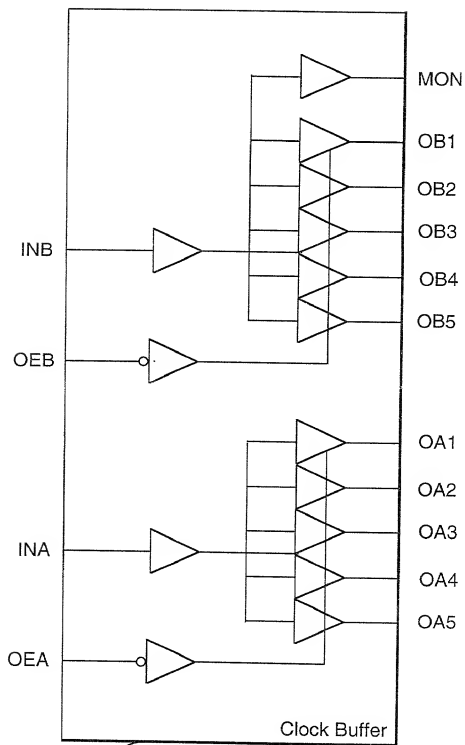
*fig. 42*

fig. 43





576

fig. 44

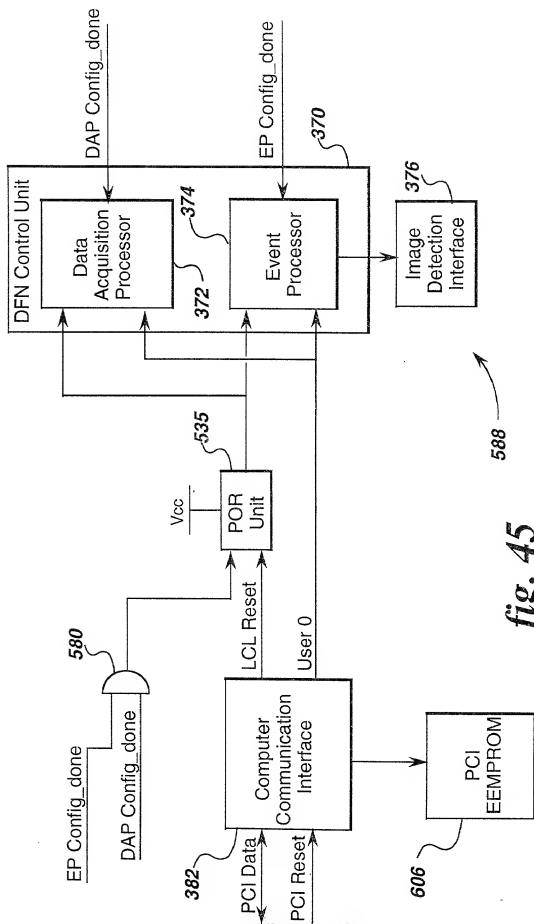


fig. 45

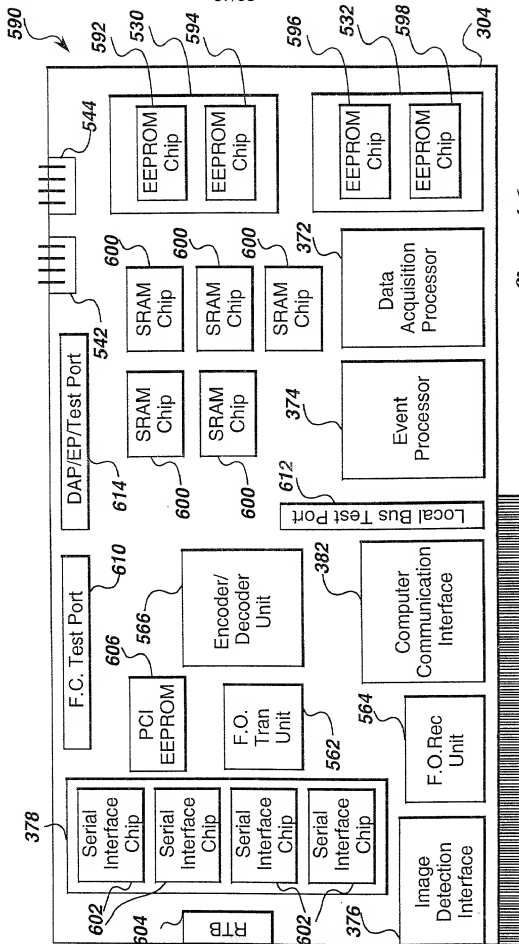
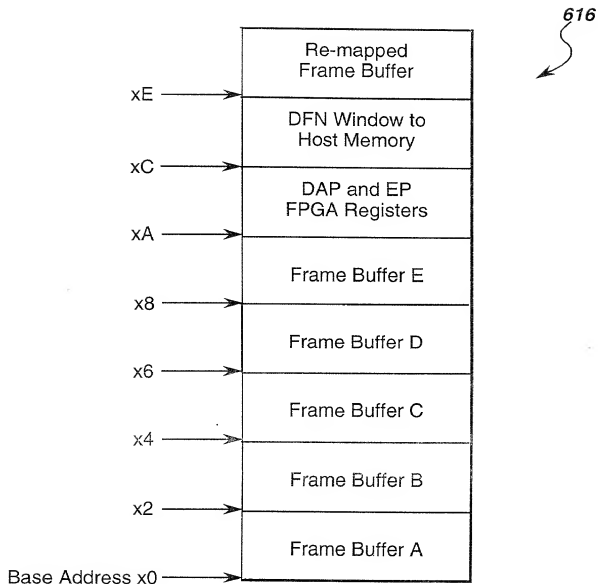
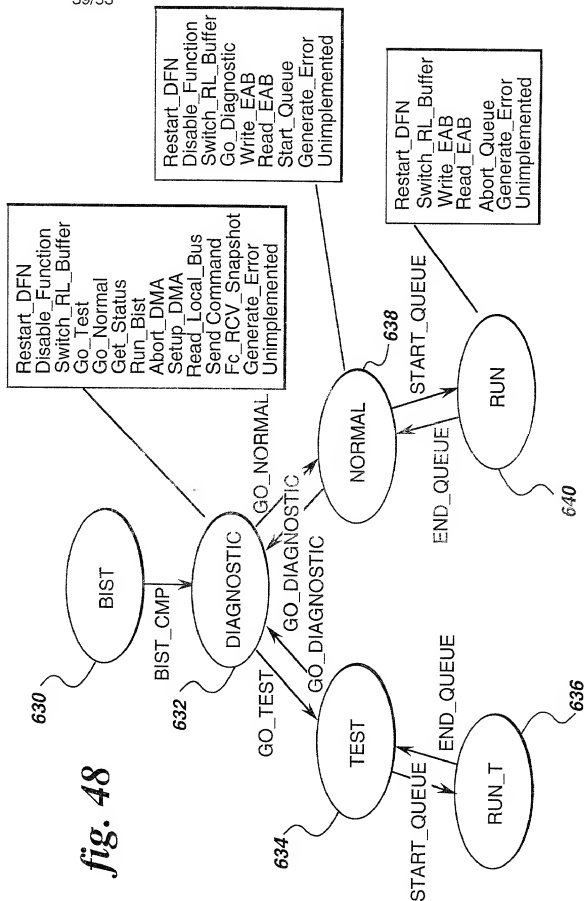


fig. 46



Mapping of 16 MByte PCI Address Space

fig. 47



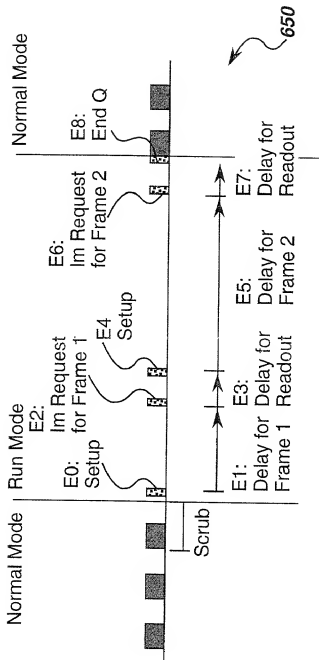


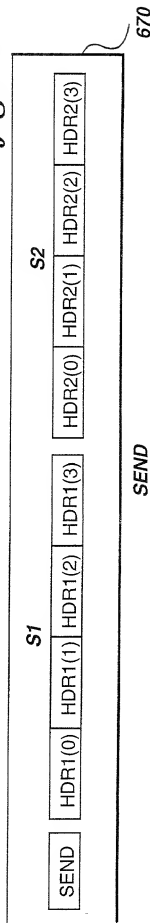
fig. 49

<i>Event Mnemonic</i>	<i>Event</i> (showing size of arguments)	<i>Op Code</i> (hex)	<i>Data</i> (bytes)	<i>Total</i> (bytes)
Endq	Endq	14	0	1
Delay (T)	Delay (0xff ff ff)	10	4	5
Send (command, value)	Send (0xff ff ff, 0xff ff ff)	04	8	9
LoopKN (K, N)	LoopKN (0xff ff, 0xff)	0C	3	4
LoopKF (K, F)	LoopKF (0xff ff, 0xff ff ff)	0D	5	6
Wait (F)	Wait (0xff ff, ff)	09	3	4
Flag (F)	Flag (0xff ff, ff)	08	3	4

660

fig. 50

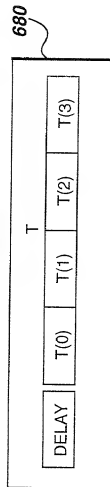
fig. 51



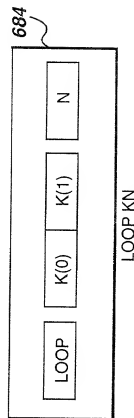
Error Mnemonic	Description of Error
FC_TIMEOUT	Timeout Expired With No ACK Detected
FC_BAD_ACK	ACK Did Not Match Transmitted Command
FC_EXTRA_ACK	Unexpected ACK Received
FC_EXTRA_CMD	New Send Event While Waiting for ACK From Previous Send
SG_DETIN	No Input Signal Power on Fibre Channel (Cable Disconnected?)
RXERROR	Fibre Channel Receiver Detected Bad Data (Defective Chipset?)
WRDSYNCN	Fibre Channel Data Link Unsynchronized
CRXS(1)	Bad Received CRC Detected (Fiber-optic Cable Problem?)
CRXS(3) & CRXS(2)	Bad Order in Link State Machine (Defective Chipset?)

672

fig. 52



DELAY T



LOOP KN

fig. 53

fig. 54

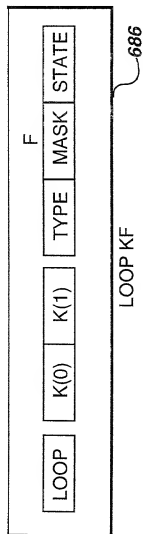


fig. 55

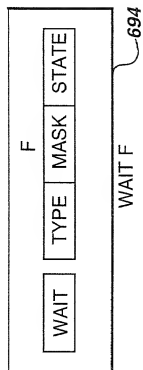


fig. 56

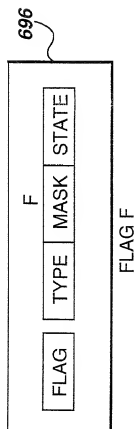


fig. 57

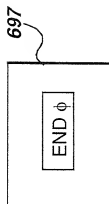


fig. 58

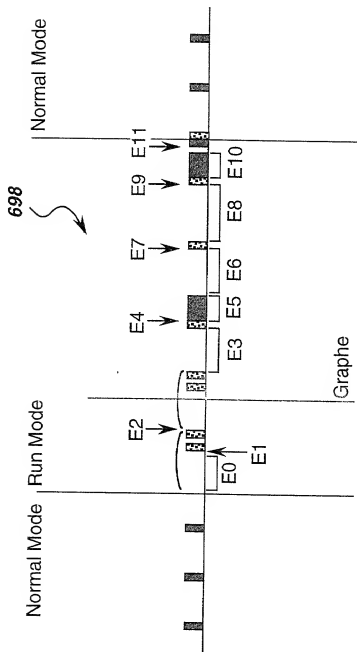


fig. 59

E11	EndQ
E10	Delay 125 ms
E9	Send Im Request
E8	Delay 500 ms
E7	Flag RT2
E6	Delay 50 ms
E5	Delay 125 ms
E4	Send Im Request
E3	Delay 300 ms
E2	Loop 2, RT1
E1	Send Scrub
E0	Delay 300 ms

EVENT QUEUE

700
↙*fig. 60*

102052-0494260

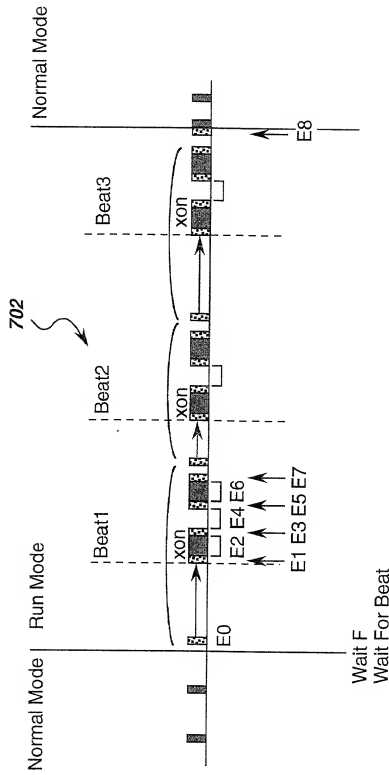
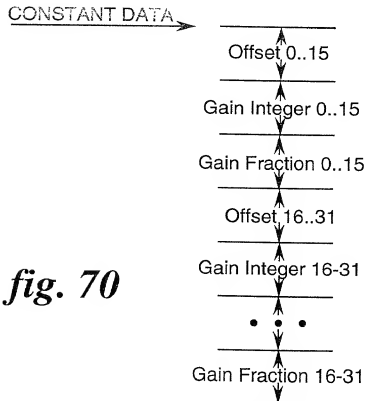


fig. 61

	EndQ
E7	LoopKF, HF1
E6	Delay 23 ms
E5	Send Im Request
E4	Delay 10 ms
E3	Flag RT2
E2	Delay 23 ms
E1	Send Scrub Request
E0	Wait RT1

EVENT QUEUE

704

fig. 62*fig. 70*

Constant Memory Format

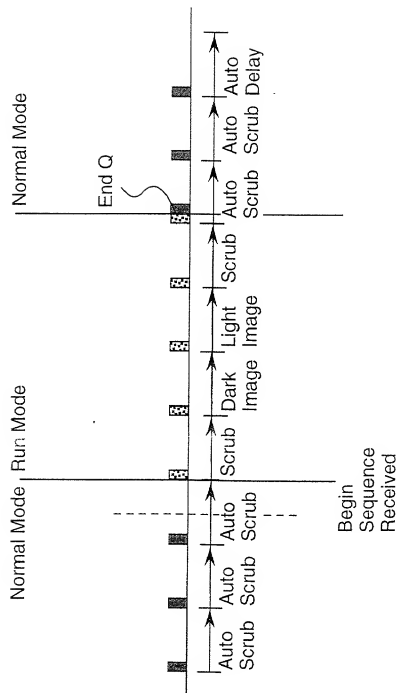


fig. 63


```
sequence_begin ( );
# define qv defaults:
%qv1 =('delay_qv' => 5000);
# call frame with qv's
frame_type1(NULL, %qv1, 1);
sequence_end ( );
```

fig. 64

```
sub :frame
{
$QVf = 'frame' ;
%qv = ('delay_qv' => [10000]);
%qp = ( );
comp:pile_init (@_, %qp, %qv, $QVf);
      Delay('Delay_qv1');
comp:pile_finit ( );
}
```

fig. 65

```
pDFN->DFNChangeQueueVariable
(
(char *) SymName,
(char *) sndBuf,
BufSize
(ULONG *) & Debug
);
```

fig. 66

```
// variable name
// new value
// num bytes to write
// developer info
```

```

// load and run the event sequence
pdfnBeginSequenceNoMappingNoLog
    (snum, "d:\\HF.bin");
// assign data to be passed
sndBuf = 25000;
// change the queue variable
pdfn->DFNChange QueueVariable
(
    (char *) SymName,           // variable name
    (char *) sndBuf,           // new value
    (ULONG) sizeof sndBuf      // num bytes to write
    (ULONG *) & debug         // developer info
);

```

fig. 67

```

sub frame_type1
{
    $HFfrm = 'frame_type1';
    %qv = ('delay_qv' => [20000]);
    %qp = ( );
    $image_cmd = [0x800000,0x0];
    compile_init (@_, %qp, %qv, $HFfrm);
    Send ($image_cmd);
    Delay('delay_qv');
    LoopKF(2, 0xAAFF01);
    compile_finit( );
}

```

fig. 68

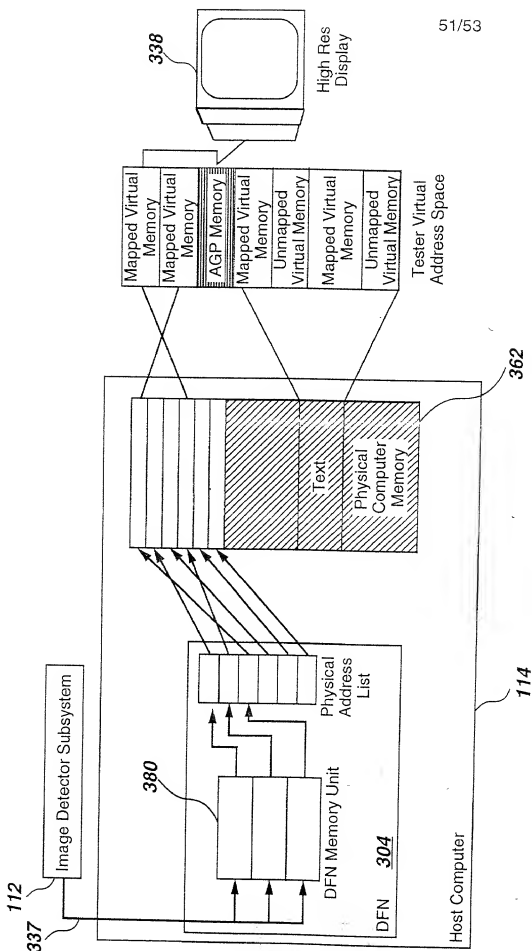


fig. 69

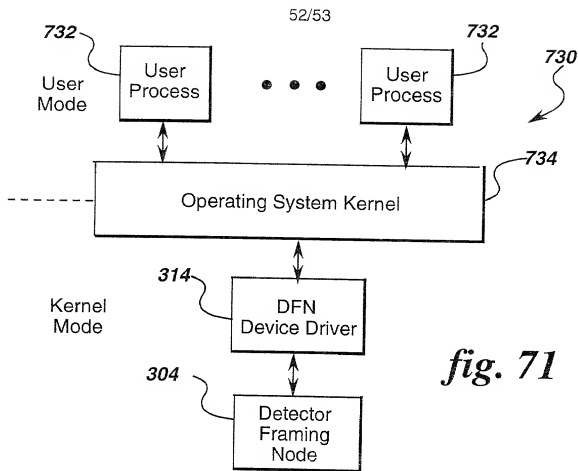


fig. 71

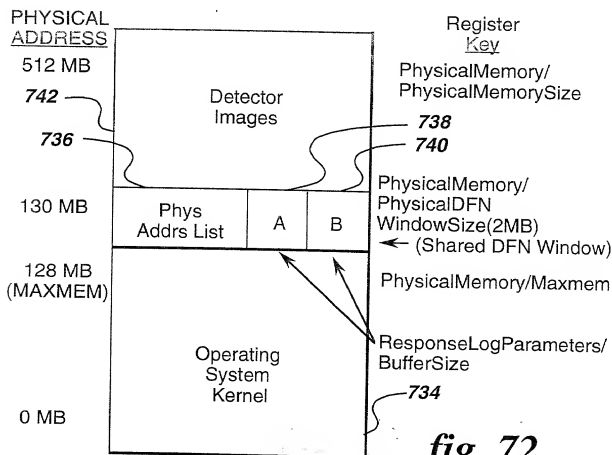
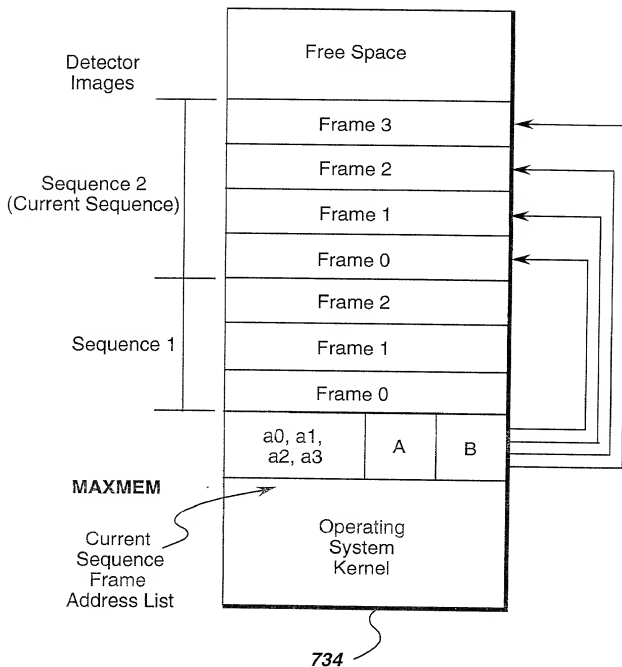


fig. 72

*fig. 73*